鹿児島大学リポジトリ
Kagoshima University
Repository

Title	ラボラトリーオートメーション
Author(s)	立野, 洋人; 岩下, 洋一朗; 深井, 晃
Citation	鹿児島大学理学部紀要. 数学・物理学・化学, 16: 71-86
Issue Date	1983-12-30
URL	http://hdl.handle.net/10232/6411

国立大学法人
鹿児島大学
KAGOSHIMA UNIVERSITY
http://ir.kagoshima-u.ac.jp

鹿児島大学理学部紀要（数学・物理学・化学），
No. 16, p. 71-86, 1983

ラボラトリーオートメーション

立野　洋人*・岩下洋一朗*・深井　晃*

（1983年9月10日受理）

Laboratory Automation

Hiroto TATENO, Youichirou IWASHITA, Akira FUKAI

(Received Sep. 30, 1983)

Abstract

Today the study of material science is so complicated and accurate, it requires amount of measurements and processings of data and rapid publication of work. To do with them, we have used microprocessor and function IC to control the measurement devices, to record and process the measured data, and to link to the data processing center.

And we have surveyed literature using Acoustic Coupler and saved effort at writing paper in English to check misspelling in paper and enable to type papers at certain format.

The result, flexibility and rapid response to the experimental condition was obtained. We wish to provide the circuit diagrams and the programs as manual.

1. はじめに

今日の物性実験は複雑化，高精度化し，大量のデータの計測と加工，及び迅速な論文の発表を要求する。これに対処するため，マイクロプロセッサ及び機能ICを使用して，計測機器の制御と計測データの収集記録及び加工，そして大型計算機とのonline化（端末機化）を行なった。

合わせて音響カプラによる文献検索及び欧論文作成の省力化を行ない，論文のミススペルのチェック及び書式指定による清書タイピングを可能にした。

この結果，実験条件に対する柔軟性及び即応性が得られたので，取扱説明書として回路図とソフトウェアを発表する。

2. CPUシステム

マイクロソフトの80系12K BASICインタプリタを自作のCPUシステムに移植し，1KのIOイニシャルプログラムを加え，13K BASICとした。ヒューズROMによるアドレスデコーダでメモリアドレス，IOアドレスの割りつけがプログラムでき，各社の80系CPUシステムをエミュレートできる。具体的にはタンディ社TRS-80モデル1のアプリケーションソフトすべてと，シャープ社MZ-80のソフトが走る。メインメモリ上のRAM, ROMの割合はボード単位で切り換えられる。低速入出力機器に対してはワンショットマルチICでウェィトをかけている。

* 鹿児島大学理学部物理学教室　Department of Physics, Faculty of Science, Kagoshima University, Kagoshima, Japan.

DOS は NEWDOS を多用している。ディスクはフルサイズからミニサイズまで自由に読み書きできる DOS である。第1図に CPU システムのブロックダイアグラム，メモリマップ，IO マップ（タンディ）を示す。

ソフトウェアとして，DOS は TRSDOS と NEWDOS があり，NEWDOS を多用している。言語は，Disk BASIC と BASIC コンパイラ, FORTRAN コンパイラ, そして MACRO80 がある。他に，EDITER, SCRIPSIT と SCRIPSIT Dictionary がある。また，ユーティリティとして RS-232C コントロールソフトと ROM 化 BASIC コンパイラ（整数型・実数型の2種類）などがある。

3. 計測・制御 CPU システム

計測・制御で他の CPU システムを製作するために ROM ライタ・ROM シミュレータ・ROM 化可能なコンパイラを製作した。

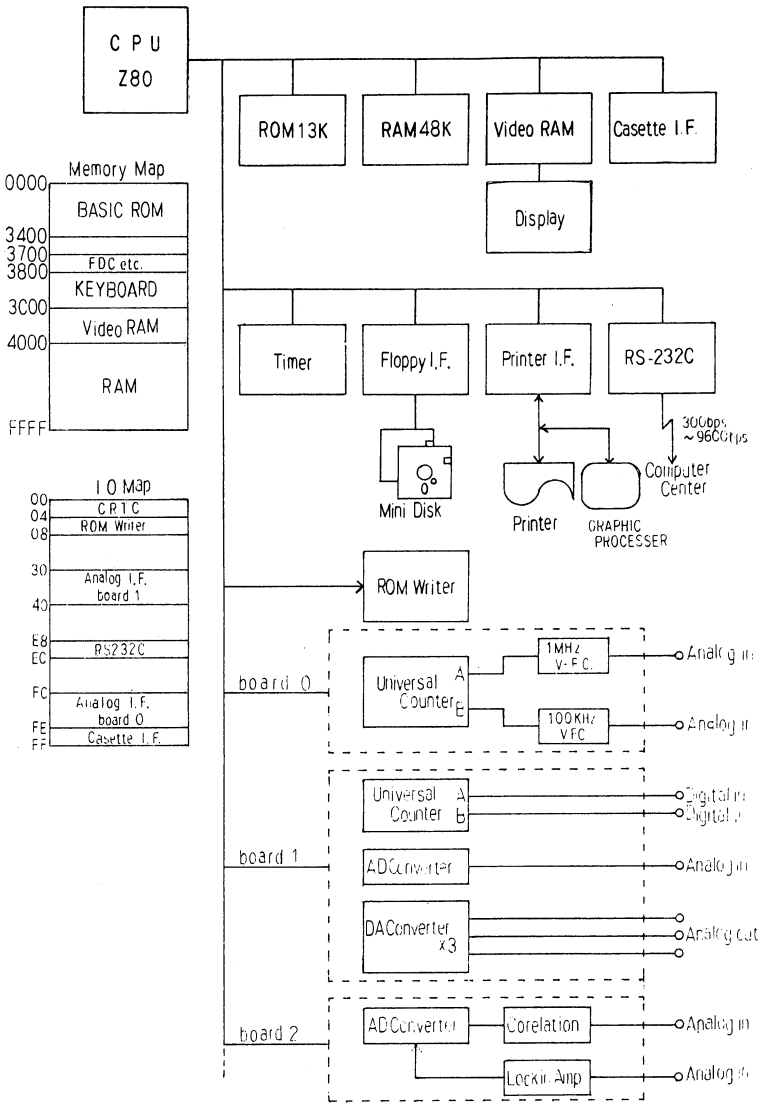

図1. CPU システムブロックダイアグラム。

(a) ハードウェア

ROM ライタは汎用入出力 LSI 8255A でマイクロコンピュータに接続した。プログラムできるメモリは EP-ROM 系（紫外線消去型）2708, 1K バイト型から，2764, 8K バイド型までの全型と，ボードを差しかえることによってプログラムできるヒューズ ROM, MB7051, MB7052 である。

EP-ROM は BASIC インタープリタ，ブートプログラム，制御プログラムなどの CPU のプログラムに用いる。

ヒューズ ROM は遅延伝播時間が短かいので TTL-IC の個別部品による論理回路をヒューズ ROM によっておきかえることができる。

プログラムのデバッグの際に，EP-ROM でプログラムすると，プログラムに時間がかかる。EP-ROM は消去に 5 分，プログラムに 5 分かかる。その為，CMOS-RAM, HM6116LP にバックアップ電池をつけて不揮発 RAM とし，これを用いて EP-ROM と同様に読み書きできる ROM シミュレータを製作した。ROM シミュレータでは，プログラムに 1 分程度しかかからない。これによりプログラムのデバッグが速くなった。第 2 図に ROM ライタの回路図を示す。ROM シミュレータは簡単なので回路図は示さない。

(b) ソフトウェア

ROM ライタの制御プログラムは BASIC で書いた。速いタイミングを要する所は BASIC プログラムから機械語サブルーチンを呼び出す様にした。EP-ROM 用のプログラムを第 3 図に示す。

プログラム開発用の言語として，当初アセンブリ言語でプログラムしていたが，ソフトウェア開発の速度・保守性を考えて，高級言語によるコンパイラを使うことにした。

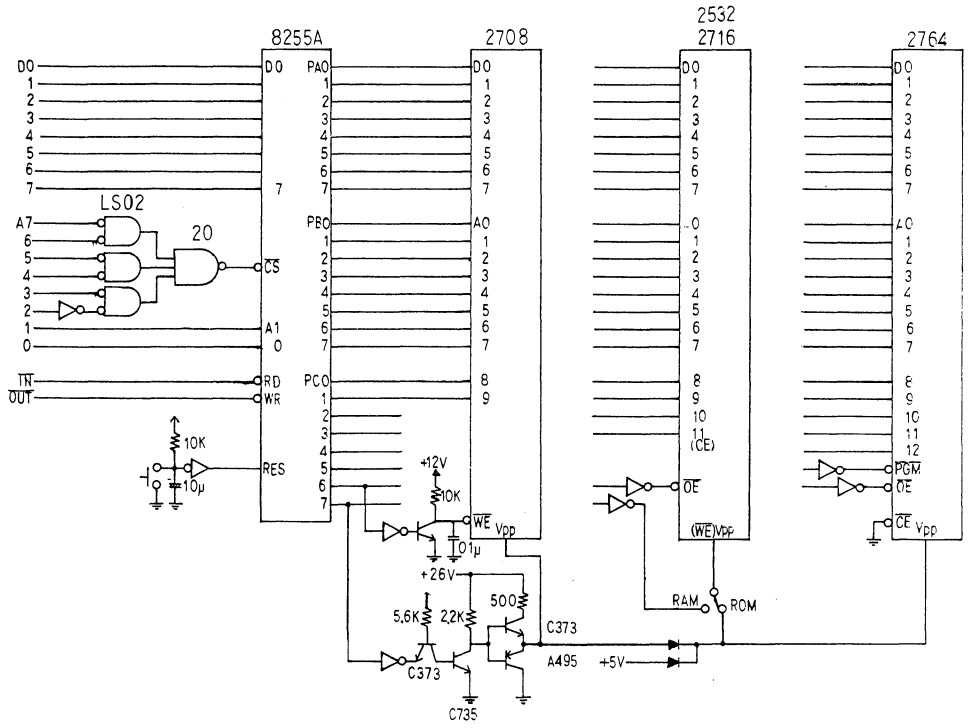

図 2. ROM ライタ回路図。

```
 5  REM****** EP-ROM WRITER CONTROL PROGRAM ******
10 DEFINT A-C,E-Z:CLS:CMD"T"
20 DEFFNHX$(A)=MID$("0123456789ABCDEF",A+1,1)
22 DEFFNHY$(A)=FNHX$(A/16)+FNHX$(AAND15)
25 DEFFNHZ$(A)=FNHY$(A/256-(A<0)*256)+FNHY$(AAND255)
30 PRINT TAB(20) "E P - R O M   W R I T E R"
40 PRINT
50 PRINT TAB(10) "WHICH ? (2708=A,2716=B,2532=C,2764=D)":PRINT TAB(16);"";
60 INPUT A$:IF A$="A" OR A$="B" OR A$="C" OR A$="D" THEN F=ASC(A$)-64 ELSE 50
65 PRINT " WAIT FOR SOME MINUITES..."
70 IF F=1 THEN PRINT "2708":FORPA=0TO(&H8F):READA$:NEXT:FOR PA=(&HE000) TO (&HE0
A1):READA$:GOSUB480:POKEPA,D:NEXT:GOTO90
80 FORPA=(&HE000)TO(&HE0BF):READA$:GOSUB480:POKEPA,D:NEXT
90 ON F GOSUB 92,94,96,98:GOTO 100
92 PRINT"2708":F=1024:G=(&HFC00):RETURN
94 PRINT"2716":F=2048:G=(&HF800):POKE(&HE032),08:POKE(&HE040),(&HF7):POKE(&HE063
),(&H07):POKE(&HE068),(&HC0):POKE(&HE085),(&H08):RETURN
96 PRINT"2532":F=4096:G=(&HF000):RETURN
98 PRINT"2764":F=8192:G=(&HE000):POKE(&HE032),32:POKE(&HE040),(&H9F):POKE(&HE063
),(&H1F):POKE(&HE068),(&HE0):POKE(&HE085),(&H20):RETURN
100 REM
110 REM EP-ROM WRITER MAIN ROUTINE
120 REM
130 INPUT"PLEASE TURN ON THE POWER SWITCH AND SW-1 OF WRITER.
  OK";A$:IF LEFT$(
A$,1)<>"Y" THEN 130
140 PA=4:PB=5:PC=6:CT=7  'PPI(8255) PORT
150 OUT CT,(&H90) 'PA=IN,PB=OUT,PC=OUT
160 OUT PC,(&H40) 'OUTPUT ENABLE
170 DEFUSR0=(&HE000)  'ERASE CHECK ROUTINE
180 DEFUSR1=(&HE019)  'WRITE SUBROUTINE
190 IF F<>1024 THEN DEFUSR2=(&HE049) ELSE DEFUSR2=(&HE052)¥) 'VERIFY SUBROUTINE
200 CLS: PRINT "IT IS EPROM WRITING PROGRAM "
210      PRINT
220      PRINT "   Using memory of 0E000H-0E0A1H"
230 PRINT
240 REM ERASE CHECK
250 INPUT "ERASE CHECK...TURN ON THE SWITCH ON THE BOARD AND
          PUT
ROM INTO SOCKET. OK";A$
260 IF LEFT$(A$,1)<>"Y" THEN 250
270 A=0
280 A=USR0(A)
290 CLS:PRINT "ERASE CHECK "
300 IF A<>F THEN PRINT : PRINT " ERROR !!  THIS ROM DOES NOT ERASED!!": PRINT "Y
OU MUST CHANGE ROM OR ERASE IT.":STOP
310 PRINT
320 PRINT " OK"
330 PRINT "NOW, WE WILL WRITE"
340 PRINT: INPUT "INPUT START ADDRESS (HEX)";A$: GOSUB 480  : IF D>32767 THEN D=
D-65536
350 D=D AND G: PRINT "ADDRESS=";FNHZ$(D)"-"FNHZ$(D+F-1): INPUT " IF OK THEN TURN
 OFF THE SWITCH ON THE BOARD. OK";A$: IF LEFT$(A$,1)="N" THEN 340
360 PRINT "WRITING....": A=USR1(D)
370 PRINT
380 PRINT "WRITE OVER"
390 PRINT "NOW, VERIFY"
410 A=USR2(D)
420 IF A<>D+F THEN PRINT : PRINT "ERROR !! WE FAIL TO WRITE!!": STOP
430 PRINT
440 PRINT "VERIFY OK"
450 PRINT "PUT OFF THE PROGRAMMED ROM."
460 INPUT "WRITE ONCE MORE";A$
470 IF LEFT$(A$,1)="Y" THEN 200 ELSE END
480 D=0:FORI=1TOLEN(A$):K=ASC(MID$(A$,I)):IFK>47ANDK<71THEND=D*16+K-48+(K>64)*7:
NEXT:RETURNELSENEXT:RETURN
1000 DATA CD,80,E0,CD,87,E0,CD,61,E0,DB,04,CD,75,E0,3C,20
1010 DATA 05,CD,70,E0,20,F0,C3,9A,0A,CD,80,E0,3E,80,D3,07
1020 DATA D3,06,7E,CD,75,E0,FE,FF,28,19,D3,04,CD,61,E0,DB
1030 DATA 06,F6,40,D3,06,01,0C,1E,0B,78,B1,20,FB,DB,06,E6
1040 DATA BF,D3,06,CD,70,E0,20,DA,C9,CD,80,E0,CD,87,E0,CD
1050 DATA 61,E0,DB,04,CD,75,E0,BE,20,BC,CD,70,E0,20,F0,18
1060 DATA B5,7C,E6,0F,47,DB,06,E6,C0,B0,D3,06,7D,D3,05,C9
1070 DATA 1B,23,7A,B3,C9,F5,4D,7C,E6,03,F6,3C,47,F1,02,C9
1080 DATA CD,7F,0A,11,00,10,C9,3E,90,D3,07,3E,40,D3,06,C9
2000 DATA CD,73,E0,CD,6A,E0,CD,91,E0,DB,04,CD,7A,E0,3C,20
2010 DATA 05,CD,8C,E0,20,F0,C3,9A,0A,CD,73,E0,3E,80,D3,07
2020 DATA E5,DD,E1,06,64,DD,E5,E1,11,00,04,7E,FE,FF,28,16
2030 DATA D3,04,CD,91,E0,DB,06,F6,80,D3,06,3E,E0,3D,20,FD
2040 DATA DB,06,E6,7F,D3,06,CD,8C,E0,20,E0,78,CD,85,E0,10
2050 DATA D4,C9,CD,73,E0,CD,6A,E0,CD,91,E0,DB,04,CD,7A,E0
2060 DATA BE,20,B3,CD,8C,E0,20,F0,18,AC,3E,90,D3,07,3E,40
2070 DATA D3,06,C9,CD,7F,0A,11,00,04,C9,F5,4D,7C,E6,03,F6
2080 DATA 3C,47,F1,02,C9,48,78,06,3C,02,41,C9,1B,23,7A,B3
2090 DATA C9,7C,E6,03,4F,DB,06,E6,40,B1,D3,06,7D,D3,5,0,0,C9
```

図3. ROM ライタコントロールプログラム。

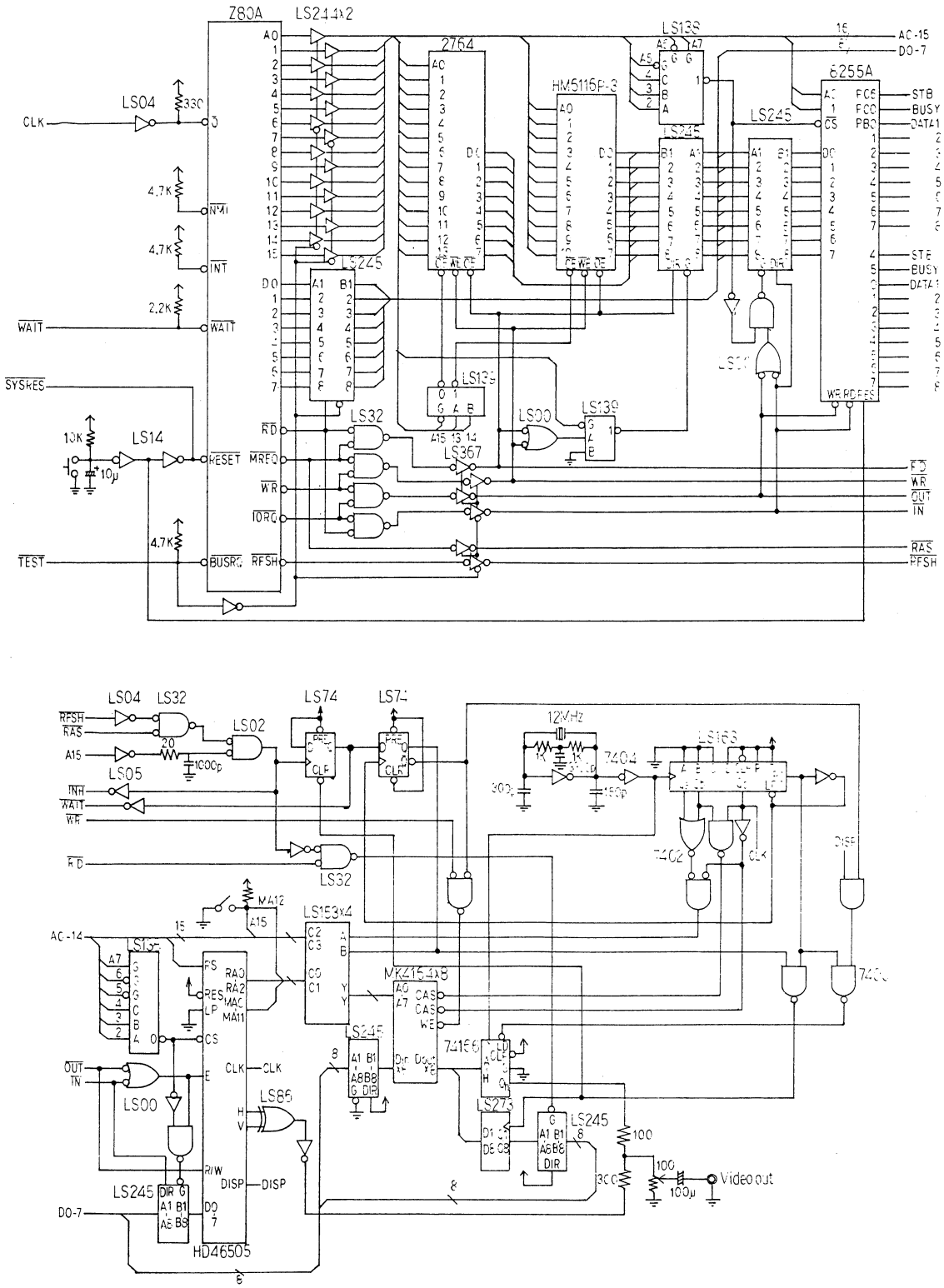

図4. グラフィックプロセッサ回路図

```
2 GOSUB 5000
5 B=&H1400:G=&H300:O=ASC("P"):H=&H1100
7 DIM H(100)
10 P=0
20 GOSUB 500:IF D=13 THEN POKE B+P,D:GOTO 50
25 IF D=9 THEN END
30 IF D<&H20 THEN 20
40 POKE B+P,D:P=P+1:GOTO 20
50 REM
100 P=0
110 D=PEEK(B+P):P=P+1
120 IF D=13 THEN RETURN
130 IF D=ASC(" ") THEN 110
140 IF D=ASC("C") THEN 1000
150 IF D=ASC("S") THEN 2000
160 IF D=ASC("R") THEN 3000
170 IF D=ASC("L") THEN 4000
180 IF D=ASC("H") THEN 6000
190 IF D=ASC("G") THEN 7000
200 GOTO 9999
500 REM*** INPUT SUB ***
510 IF (INP(&H86)AND32)=0 THEN 500
520 D=INP(&H84):RETURN
1000 D=0
1010 FOR A=&H8000 TO &HFFFF STEP 2
1020 POKE A,D:POKE A+1,D
1030 NEXT A
1040 GOTO 9999
2000 GOSUB 2500:X=D
2010 GOSUB 2500:Y=D
2020 C=ASC("P")
2030 IF O=ASC("O") THEN 9999
2045 GOSUB 3500
2050 GOTO 2000
2500 D=PEEK(B+P)
2505 IF D=13 THEN O=ASC("O"):RETURN
2510 IF D=ASC(" ") THEN P=P+1:GOTO 2500
2520 IF D=ASC("-") THEN S=-1:P=P+1:GOTO 2540
2530 S=1
2540 D=0
2550 N=PEEK(B+P)-ASC("0")
2560 IF N<0 OR N>9 THEN 2600
2570 D=D*10+N:P=P+1
2580 IF P>G THEN O=ASC("O"):RETURN
2590 GOTO 2550
2600 REM
2610 D=D*S:RETURN
3000 GOSUB 2500:X=D
3010 GOSUB 2500:Y=D
3020 C=ASC("R")
3030 IF O=ASC("O") THEN 9999
3045 GOSUB 3500
3050 GOTO 3000
3500 IF X<0 OR X>511 OR Y<0 OR Y>479 THEN RETURN
3505 A=INT(X/8)+(Y*64)+&H8000
3510 D=1:IF (X AND 7)>0 THEN FOR I=1 TO (X AND 7):D=D*2:NEXT I
3520 IF C=ASC("P") THEN D=D OR PEEK(A):GOTO 3600
3530 IF C=ASC("R") THEN D=D XOR 255 AND PEEK(A):GOTO 3600
3540 GOTO 3610
3600 POKE A,D
3610 RETURN
4000 GOSUB 2500:J=D
4010 GOSUB 2500:K=D
4020 IF O=ASC("O") THEN 9999
4030 GOSUB 2500:L=D
4040 GOSUB 2500:M=D
4050 IF O=ASC("O") THEN 9999
4060 C=ASC("P")
4075 GOSUB 4500
4080 J=L:K=M:GOTO 4030
4500 F=0:U=ABS(L-J):V=ABS(M-K)
4510 W=(L-J>0)-(L-J<0) Z=(M-K>0)-(M-K<0)
4520 IF U>=V THEN 4550
4530 F=-1:T=J:J=K:K=T:T=L:L=M:M=T
```

```
4540 T=U:U=V:V=T:T=W:W=Z:Z=T
4550 R=U/2
4560 IF F=0 THEN X=J:Y=K:GOTO 4580
4570 X=K:Y=J
4580 GOSUB 3500
4590 IF J=L THEN 4630
4600 J=J+W:R=R+V
4610 IF R<U THEN  4560
4620 R=R-U:K=K+Z:GOTO 4560
4630 IF F=-1 THEN T=L:L=M:M=T
4640 RETURN
5000 A=128:D=129
5010 OUT A,0:OUT D,95:OUT A,1:OUT D,64:OUT A,2:OUT D,77:OUT A,3:OUT D,4
5020 OUT A,4:OUT D,34:OUT A,5:OUT D,4:OUT A,6:OUT D,30:OUT A,7:OUT D,32
5030 OUT A,8:OUT D,3:OUT A,9:OUT D,7:OUT A,10:OUT D,0:OUT A,11:OUT D,0
5040 OUT A,12:OUT D,0:OUT A,13:OUT D,0:OUT A,14:OUT D,0:OUT A,15:OUT D,0
5050 FOR A=&H8000 TO &HFFFF:POKE A,0:NEXT A
5060 RETURN
6000 D=13:GOSUB 6500:D=8:GOSUB 6500
6010 FOR L=0 TO 73:FOR J=0 TO 479:K=479-J
6020  A=(L*7/8)+(K*64)+&H8000:E=PEEK(A):F=PEEK(A+1)
6030  M=L*7:M=M-INT(M/8)*8
6040  IF M=0 THEN D=E:GOTO 6080
6050  FOR I=1 TO M:E=E/2:NEXT I
6060  FOR I=1 TO 8-M:F=F*2:NEXT I
6070  D=E OR F
6080  D=D AND &H7F OR &H80:GOSUB 6500
6090 NEXT J:NEXT L
6100 D=13:GOSUB 6500
6110 GOTO 9999
6500 REM** OUT SUB **
6510 IF (INP(&H86)AND1)=1 THEN 6510
6520 OUT &H85,D
6530 OUT &H87,&H0C:OUT &H87,&H0D
6540 RETURN
7000 REM*** GRAPH PLOT ***
7010 GOSUB 2500:H(0)=D:E=D
7020 GOSUB 2500:H(1)=D:Q=D
7030 I=2
7040 GOSUB 2500:H(I)=D:I=I+1
7050 IF D<E THEN E=D
7060 GOSUB 2500:H(I)=D:I=I+1
7070 IF D<Q THEN Q=D
7080 IF O=ASC("O") THEN L=I-3:GOTO 7100
7090 GOTO 7040
7100 FOR I=0 TO 479:A=I*64+&H8000
7110 POKE A,PEEK(A)OR 1
7120 IF INT(I/10)*10=I THEN POKE A,PEEK(A) OR &H1F
7130 NEXT I
7140 FOR A=&HF7C0 TO &HF7FF STEP 2:POKE A,&HFF:POKE A+1,&HFF:NEXT A
7150 FOR X=10 TO 510 STEP 10:FOR Y=475 TO 479:GOSUB 3500
7160 NEXT Y:NEXT X
7170 C=ASC("P"):FOR N=0 TO L STEP 2
7180  J=H(N)-E K=479-H(N+1)+Q
7190  FOR M=-2 TO 2
7200   X=J+M:Y=K+M:GOSUB 3500
7210   X=J-M:GOSUB 3500
7220  NEXT M
7230 NEXT N
7240 GOTO 9999
9999 GOTO 5
```

図5. グラフィックプロセッサコントロールプログラム。

コンパイラの特徴は，プログラムが書きやすく，算術演算は一般的な数式による表記ができ，扱える数値，演算子もアセンブラより多いこと，またコンパイルの段階で不合理な演算，不適当なプログラム構造がチェックされ，コーディングミスがソースレベルで発見できること，細かな演算操作に気を配る必要が少なく大局的なプログラミングができることなどである。

ここではROM化可能なBASICコンパイラを開発してターゲットCPUプログラムのバイナリーオブジェクトプログラムを開発している。

この整数型BASICは8Kバイトで，その扱える変数の範囲は−32768～32767の整数で，変数は26個あり，配列は一次元まで，算術演算は四則演算が可能である。ここではコンパイラのプログラムは示さない。このコンパイラの使用例は後で示す。

他に実数も扱えるROM化可能なBASICコンパイラも開発した。実数演算が必要な実験の場合は，このコンパイラを使う。

4. データ表示

測定データ及び処理データのグラフ表示の為にインテリジェントグラフィックプロセッサを製作した。CPUはメインCPUからの命令，データを受け取り，解釈，実行，表示する。メインCPUからデータを受け取り，最大，最小値を取り，座標軸を描いて，ビデオディスプレイに表示させてグラフを表示することができる。

CPUにはZ80A, CRTコントローラにはHD46505SP，ビデオメモリには64KビットダイナミックRAM MK4164Sを用い，512×480ドットの白黒画面を2枚持ち，スイッチで2枚の画

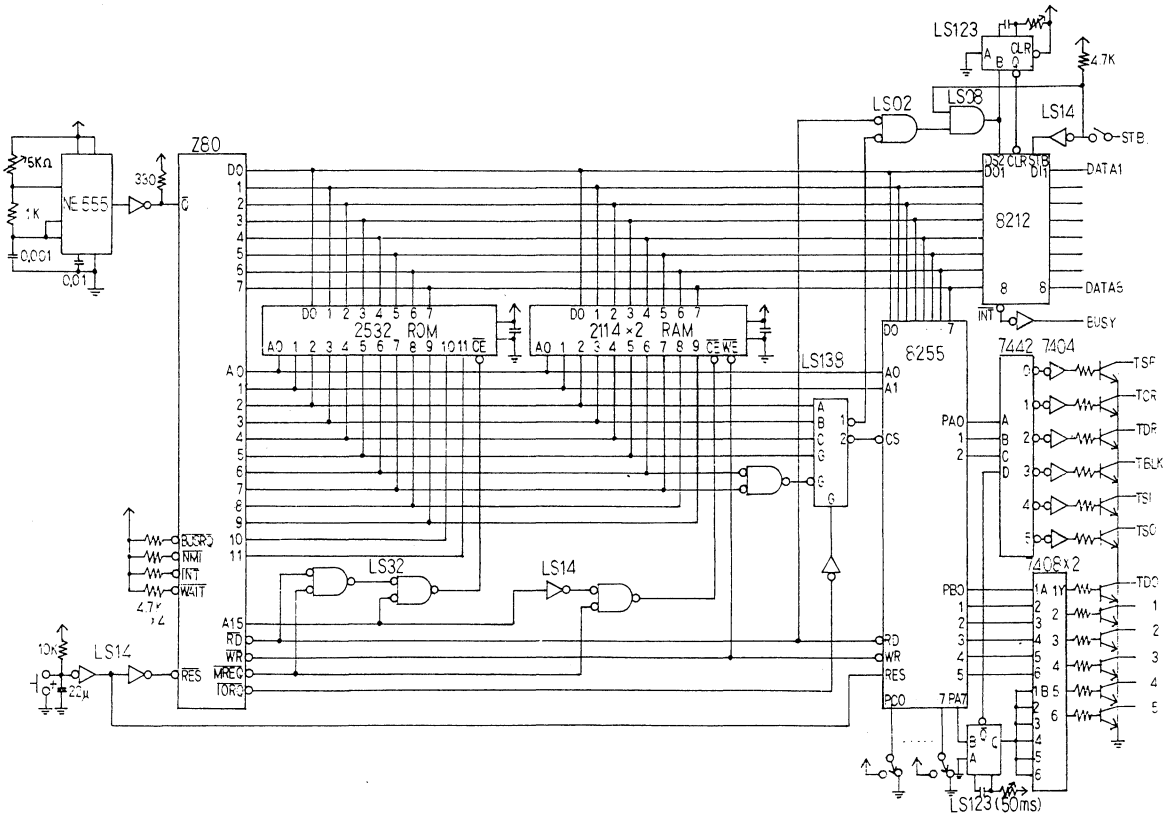

図6.　タイプインターフェース回路図。

面のどちらかの表示を切換えられる。またCOPY命令により，その画面のハードコピーが出力できる。

プログラムには前述のROM化コンパイラを用いてプログラムの開発を行った。回路図とプログラムリストを第4図，第5図に示す。

5. 論文作成

ラボラトリオートメーションの一環として論文作成の省力化を試みた。

(a)ソフトウェア

物理実験は次々に高性能の装置や新しいシステムなどが導入され，進歩するのに対して，論文作成，文献検索などの文書処理は，情報は増え続けるのに処理する方法はなかなか進歩せず，処理に時間と労力を消費する。単に不慣れな外国語ということ以上に，原稿のタイピングから修正，打ち直し，再修正，仕上げといった一連のテキストプロセシングの過程は極めて煩雑なものがあり，多大の労力と時間を必要とする。

英文ワードプロセッサはタイプしたい文章を見やすいビデオディスプレイで確認しながらタイプインでき，一度タイプインした文章を，画面を見ながら誤字の修正をしたり，新たに語句の挿入や削除が強力なスクリーンエディット機能で簡単に出来る。通常のタイプライタの様に

```
5 REM **** TYPEWRITER INTERFACE CONTROL PROGRAM ****
7 REM **********TYPECTRL/BAS***********
10 A=&H700:B=&H8000:F=0
15 OUT 4,0:OUT 11,&H82
20 FOR J=0 TO 255
30  POKE B+J,PEEK(A+J):NEXT J
40 IF (INP(9)AND1)=1 THEN GOSUB 500
65 K=130:GOTO 900
70 I=INP(4)
72 IF I=35 THEN 400
73 P=INP(9)
76 IF P>127 THEN 73
80 C=PEEK(B+I)
90 IF C=0 THEN 70
95 IF C=1 THEN 70
100 IF C<63 THEN 300
110 IF C>200 THEN 200
120 IF F=0 THEN GOSUB 700
130 GOTO 300
200 IF F=1 THEN GOSUB 800
300 IF C=13 THEN 900
310 IF K>130 THEN 900
320 OUT 8,0:OUT 8,C OR 128:K=K+1
330 GOSUB 600
340 GOTO 70
400 IF (INP(9)AND64)=64 THEN 400
410 IF (INP(9)AND32)=32 THEN 80
420 GOTO 70
500 FOR J=0 TO 255
510 I=INP(4):IF I=0 THEN 510
515 POKE B+J,I
520 NEXT J:RETURN
600 M=INP(9):M=M AND 15:M=M+1
610 FOR L=0 TO M*160:NEXT L:RETURN
700 OUT 8,0:OUT 8,142:F=1
710 FOR N=0 TO 200:NEXT N:RETURN
800 OUT 8,0:OUT 8,143
810 F=0:N=0
820 FOR N=0 TO 200:NEXT N:RETURN
900 OUT 8,0:OUT 8,141
910 FOR N=0 TO 200:NEXT N
920 FOR L=0 TO K*16:NEXT L
930 K=0:GOTO 70
```

図7. タイプインターフェースコントロールプログラム。

文章の追加・削除があった場合に全文を打ち直す必要は全くない。またプリンタの出力フォーマットを自由に指定でき，右揃え，左揃え，左右揃え，センター揃えなどが簡単にできる。たとえば，日本物理学会（J.P.S）の欧論文投稿規定65字27行の打ち出しが簡単に出来る。タイプインした文章をディスクにファイルしておくことができ，後で必要な時に読み出してスピーディに文書作成ができる。また高品質に印字するためIBMタイプライタを使うことによって美しいプリントアウトができ，印刷原稿にも使える。

スペリングチェックのために，ミニディスケットに七万二千語の英単語のはいった電子辞書を求め，ワードプロセッサで作った文章ファイルを校正している。専門用語は二千四十五語追加登録できる。また，ワードプロセッサの強力なエディット機能でBASICやFORTRANのプログラムが作れる。電子辞書に各々の言語のコマンドを登録しておくと，ミスインプットがチェックできる。

(b)ハードウェア

論文打ち出しのプリンタとしてIBMの電動タイプライターをCPUにインターフェースした。

接続はセントロニクス規格とし，インテリジェント機能があり，IBMゴルフボールに無い文字が来たら停止し，たとえばギリシャ文字のゴルフボールと取り変えられる。数式を肉筆で書きこまれなければならない場所も印字しないストップコード，たとえば"#"を前もって文章に挿入しておくと，検出してタイプライターはストップする。

IBMタイプライターには直結タイプとソレノイドプランジャーでキーボードをたたく間接タイプがあり，両タイプともインテリジェント機能のあるインターフェースを作った。その回路図とリストを第6図，第7図に示す。

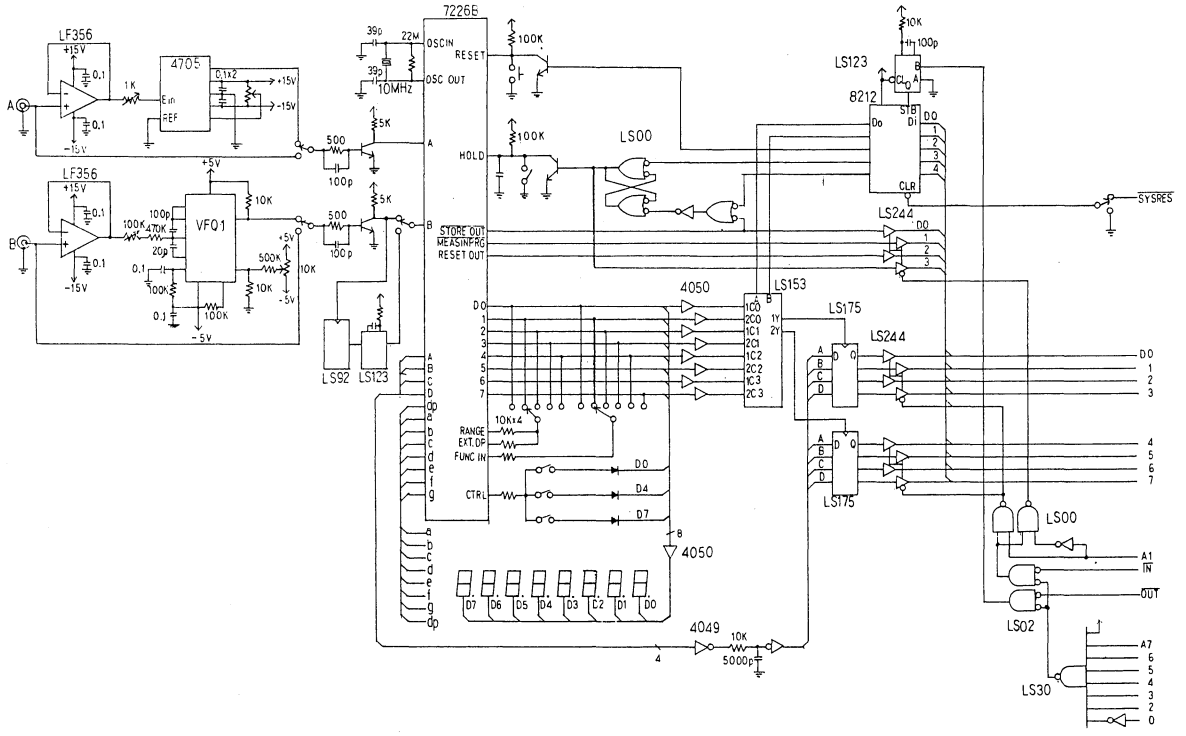

図8.　アナログインターフェース回路図ボード1

6. 計測器とのインターフェース

マイクロコンピュータシステムと外部計測制御機器とのインターフェースは，コンピュータで扱えるデジタル信号と，計測制御機器で扱われるアナログ信号の変換を行なう。

ボード0では，アナログ信号は電圧の形で入力され，V-Fコンバータ（電圧-周波数変換器）を通して周波数になる。周波数を8桁ユニバーサルカウンタでカウントする。V-Fコンバータは最大1 MHzであり，10Vが，1秒で6桁，100秒で8桁の精度で得られる。また，もう一つV-Fコンバータがあり，ユニバーサルカウンタの機能により，2つの電圧の比が得られる。V-Fコンバータとカウンタの組み合わせにより，積算して情報量を下げずに雑音を軽減できる。

ボード1は，12ビットのA-DコンバータとD-Aコンバータが接続される。D-Aコンバータは制御機器に参照信号を送る他にX-tレコーダに即時処理したデータを送って記録させている。A-Dコンバータは電圧を直接CPUに入力する。また，ユニバーサルカウンタは，8桁の精度でカウントする。さらに，周波数シンセサイザを接続して，1 Hz～200 kHzまで1 Hzきざみで歪率3%以内の正弦波を出力できる。他にアナログレコーダ出力(X-tあるいはX-Yレコーダの出力）をデジタライザでCPUに入力できる。

以上はすべて汎用パラレルインターフェースLSI 8255Aで接続され，データバスはショットキーTTLバッファで特別に強化した。

他に自己相関計やデジタルロックインアンプがある。

以上の回路図を第8図，第9図を示す。

7. 応　用　例

ボード0, 1と2を使用した内部摩擦（IF）測定ソフトはメニュー形式で実験条件はキーボードから指定できる。ソフトはBASICで書かれている。IFの測定装置はアナログ的に自動制御化されたマルクス複合振動子法である。試料は水晶振動子で周期的応力振幅V_dが加えられ，歪振幅V_gが生じ，IFはV_d/V_gに比例する。ヤング率はこの時の共鳴周波数fとして測定される。

CPUシステムからは参照歪振幅V_Rがアナログ制御系へ与えられ，振幅依存性のIF，ヤング率が測定できる。

V_d, V_gはボード中でVFコンバータにより周波数に変換され，8桁のユニバーサルカウンタで周波数比として測定される。これは時間をかければ8桁の測定精度が得られることを示す。fはボード1で読み取られ，CPUでそれぞれ実験式の処理をし，ボード1のDAコンバータから3ペンX-tレコーダに内部摩擦，ヤング率，歪振巾がそれぞれ出力される。

計測データはすべてフロッピーディスクにファイルされ，同時にグラフィックプロセッサ及びプリンタに出力される。

IFの時間依存性の測定ソフトと，グラフィック出力をそれぞれ第10図，第11図に示す。

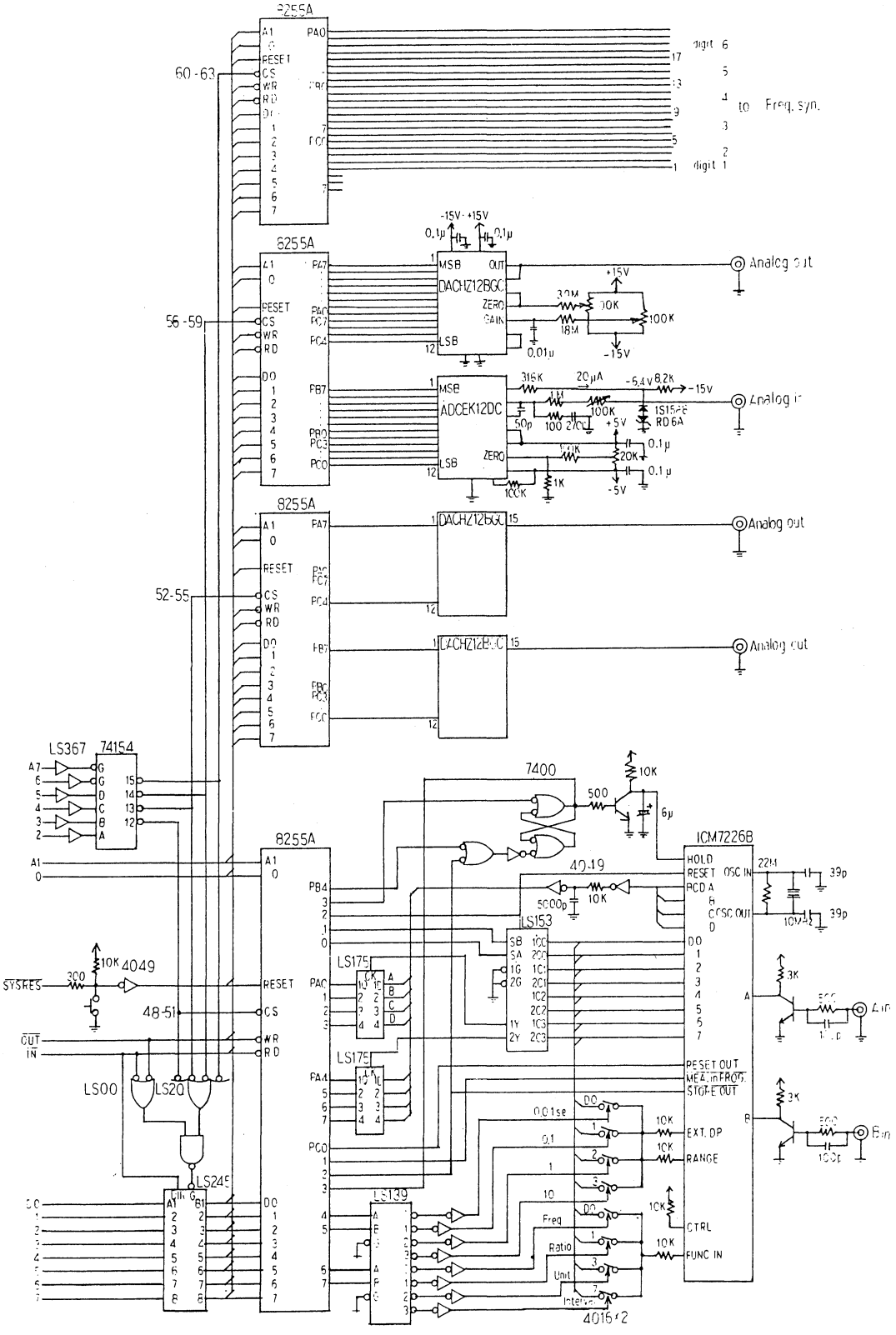

図 9.　アナログインターフェース回路図ボード 1

```
10   '****** Internal Friction Measurement Program ******
20   '**********************written by Y.Iwa.**************
30   '**********************07/19/83**********************
40   '************INTFRIC2/BAS*****************************
50  'INITIALIZE
60   OUT 51,145 'Freq. Counter(5)
70   OUT 55,128 'DAC(3),(4)
80   OUT 59,131 'DAC(1),ADC(2)
90   OUT 63,128 'SYNTHESIZER
100  OUT 50, 48:GOTO 190  '10 SEC,RATIO
110 OUT 56,NOT(D/16)AND255:OUT 58,(NOT D AND 15)*16 OR (INP(58)AND15):D1=D: RETU
RN
120 DS=D1:RETURN
130 RETURN
140 DS=FND2(P):RETURN
150 OUT 52,NOT(D/16)AND255:OUT 54,(NOT D AND 15)*16 OR (NOT D4 AND 15):D3=D:RETU
RN
160 DS=D3:RETURN
170 OUT 53,NOT(D/16)AND255:OUT 54,(NOT D3 AND 15)*16 OR (NOT D AND 15):D4=D:RETU
RN
180 DS=D4:RETURN
190 D=0:GOSUB110  :GOSUB150  :GOSUB170
200 CLEAR 2000:DEFINT A-Z
210 C=0:D=0:D1=0:D4=0:DI=0:DE=0:DS=0:EN=4095:I=0:J=0:MODE=0:N=0:P=0:ST=0:VD=0:VS
=10:W=0:W1=0:J1=0:J2=0:T=0:T0=0:TS=0:A=(&H4041)
220 DF#=0:DE#=0:KE#=1:KF#=1:FI#=0:DG#=0:DD#=0:DO#=0:OF#=0:DP#=0
230 DEF FND1(A)=(NOTINP(56)AND255)*16+(NOTINP(58)AND240)/16
240 DEF FND2(A)=(INP(57)AND240)*100/16+(INP(57)AND15)*10+(INP(58)AND15)
250 DEF FND3(A)=(NOTINP(52)AND255)*16+(NOTINP(54)AND240)/16
260 DEF FND4(A)=(NOTINP(53)AND255)*16+(NOTINP(54)AND15)
270 U$=CHR$(&H21)+CHR$(&HC0)+CHR$(&H3C)+CHR$(&H11)+CHR$(&H80)+CHR$(&H3C)+CHR$(&H
01)+CHR$(&H40)+CHR$(&H03)+CHR$(&HED)+CHR$(&HB0)+CHR$(&HC9)
280 UA=VARPTR(U$):U!=PEEK(UA+1)+PEEK(UA+2)*256
290 U!=U!+65536*(U!>32768)
300 DEFUSR0=U!
310 L0$="************* INTERNAL FRICTION MEASUREMENT SYSTEM ************"
320 CM$="":F$="####":G$=STRING$(4,8)+F$:A$="":I$="":L$=""
330 L1$="DAC(1)    ADC(2)    DAC(3)    DAC(4)    FREQ.(5)      VOLTDIV.(6)    "
340 L2$="   ####     ####       ####      ####     ########       ########       ##.###
V  #.## V    ##.###V  ##.###V                                  "
350 L3$=" Vg(3)  Vd/Vg(6)   d(Vd/Vg)  Rec(4)  Freq.(5)  ADC(2) I.F.(1)   "
360 L4$=" ####    ########   ########   ####   ########  ####      ####    "
370 N=1500: DIM DE#(N),FR#(N),AD(N)
380 CLS
390 PRINT @ 0,L0$;
400 MODE=0:GOSUB 870  : MODE=1:I$=""
410 PRINT @ 64,L1$;
420 PRINT USING L2$;FND1(A),FND2(A),FND3(A),FND4(A),DF#,DE#,FND1(A)*10/4096,FND2
(A)/100,FND3(A)*10/4096,FND4(A)*10/4096;
430 PRINT "I/O Port Number            ";CHR$(31)
440 PRINT "DAC(1)=1, ADC(2)=2, DAC(3)=3, DAC(4)=4, Freq.(5)=5, Volt(6)=6,"
450 PRINT "Go measurement=10  ";
460 PRINT: PRINT "WHICH(1-10)<";P;">";:INPUT P
470 IF P<1 OR P>10 THEN END
480 ON P GOSUB 500  ,620  ,500  ,500  ,870  ,870  ,390  ,390  ,390  ,1000
490 GOTO 390
500 '--DAC(1,3,4)--
510 PRINT USING "DAC(#) VOLT";P;:INPUT DE
520 ON P GOSUB 120  ,140  ,160  ,180
530 IF DE<0 OR DE>4095 THEN RETURN
540 IF MODE AND 2 THEN PRINT USING F$;DS;
550 FOR D=DS TO DE STEP SGN(DE-DS)
560   ON P GOSUB 110  ,130  ,150  ,170
570   IF MODE AND 1 THEN PRINT @ 120+9*P, USING F$;D;
580   IF MODE AND 2 THEN PRINT USING G$;D;
590   IF I$="" THEN I$=INKEY$ ELSE D=DE
600 NEXT
610 RETURN
620 '--ADC(2)---
630 PRINT USING "ADC(2)=#.## V    ";FND2(P)/100;
640 IF INKEY$="" THEN 630
650 RETURN
```

```
660 '-- Freq. Counter --
670 IF P=5 THEN IF (INP(50) AND 8)=8 THEN IF I$="" THEN I$=INKEY$: GOTO 670  ELS
E 750
680 DF#=0
690 FOR J=3 TO 0 STEP -1
700   OUT 49,J+28:FOR W=1 TO 10: NEXT 'WAIT
710   DI=INP(48)
720   DF#=DF#*100+INT(DI/16)*10+(DI AND 15)
730 NEXT
740 IF MODE AND 1  THEN PRINT "Frequency=";DF#,: IF INKEY$="" THEN 870
750 RETURN
760 '-- Voltage divider ,-
770 IF (INP(252) AND 8)=8 THEN IF I$="" THEN I$=INKEY$: GOTO 770  ELSE 860
780 DE#=0
790 FOR J=3 TO 0 STEP -1
800   OUT 254,J+8+16
810   FOR W=1 TO 10:NEXT 'WAIT
820   DI=INP(254)
830   DE#=DE#*100+INT(DI/16)*10+(DI AND 15)
840 NEXT J
850 IF MODE AND 1  THEN PRINT " Vd/Vg  ratio=";DE#,: IF INKEY$="" THEN 870
860 RETURN
870 '-- DOUBLE COUNTER READ --
880 IF P=5 THEN OUT 49,20: OUT 49,28 'Reset pulse of Freq.
890 OUT 254,16:OUT254,24 'Reset pulse of Div.
900 IF P<>6 THEN GOSUB 670  'Freq.
910 IF P<>5 THEN GOSUB 770  'Div.
920 RETURN
930 '*** MODE means.....
940 '   MODE = 1... Display read DATA of FREQ.(5) or RATIO(6)
950 '          2... Display output DATA of DAC(1,3,4)
960 '          4... Record DATA to Arrays DE#(I),FR#(I),AD(I)
970 '          8... PRINT OUT TO PRINTER in Measurement
980 '          16.. PRINT OUT TO PRINTER at regular intervals
990 '                                          in measurement
1000 '-- Measurement ---
1010 PRINT "START VOLT(0-4095),<";ST;">";:INPUT ST:IF ST<0 THEN RETURN
1020 P=3:MODE=2:DE=ST:PRINT CHR$(27);STRING$(32,25);:GOSUB 520  :PRINT
1030 PRINT "END   VOLT(0-4095),<";EN;">";:INPUT EN
1040 PRINT "STEP  VOLT(1-4095),<";VS;">";:INPUT VS:'IF VS=<0 THEN 1040
1050 PRINT "DAC(4) ｶﾗ X-t Recorder ｴ ﾀﾞｽ ﾊﾞｲﾘﾂ d(Vd/Vg)/N,N=<";KE#;">";:INPUT KE
#
1070 PRINT "DAC(1) ｶﾗ X-T RECORDER ｴ ﾀﾞｽ ﾊﾞｲﾘﾂ VD/VG/N,   N=<";KF#;">";:INPUT KF
#
1075 PRINT "Vd/Vg ﾉ ｵﾌｾｯﾄ <";OF#;">";:INPUT OF#
1085 IF VS=0 THEN MODE=0 :GOTO 1100
1090 A$="N":IF N>ABS(EN-ST)/VS THEN PRINT "Remember DATA (Y/N),<";A$;">";:INPUT
A$: IF A$="Y" THEN MODE=4 ELSE MODE=0 ELSE MODE=0
1100 T0=0:T=0:I=0 :I$=""'ﾊｲﾚﾂ, ｻｲｼｮ ﾉ 5 ﾌﾝ ﾊ ﾏｲｶｲ ﾄﾙ
1110 C=2 'CURSOR
1120 PRINT "PRINTER OUT(Y/N),<";A$;">";:INPUT A$: IF A$="Y" THEN MODE=MODE OR 8
1125 PRINT "INTERVAL PRINT OUT(Y/N),<";A$;">";:INPUT A$:IF A$="Y" THEN MODE=MODE
 OR 16:PRINT "INVERVAL TIME (sec),<";TS;">";:INPUT TS
1127 PRINT "INTERVAL DISK OUT (Y/N),<"A$">";:INPUT A$:IF A$="Y" THEN MODE=MODE O
R 32:PRINT "INTERVAL TIME(sec),<"TS">";:INPUT TS:LINE INPUT "File Name ? ";L$:OP
EN "O",1,L$
1130 IF MODE AND 24 THEN LINEINPUT "Comment ? ";CM$: LPRINT CM$
1135 IF MODE AND 32 THEN LINEINPUT "COMMENT ? ";CM$:PRINT#1,CM$
1140 PRINT "START OK(Y/N),<";A$;">";:INPUT A$
1150 IF A$<>"Y" THEN 1000
1160 PRINT @ 0, L0$;:PRINT @ 64, L3$;CHR$(31);
1165 IF MODE AND 32 THEN PRINT @3,L$;
1170 IF MODE AND 8 THEN LPRINT L3$
1175 IF MODE AND 16 THEN LPRINT "TIME(sec) ";L3$:CMD"T":POKEA-1,0:POKEA,0:POKEA+
1,0:POKEA+2,0:CMD"R"
1177 IF MODE AND 32 THEN CMD"T":POKE A-1,0:POKE A,0:POKE A+1,0:POKE A+2,0:CMD"R"
1180 FOR VD=ST TO EN STEP VS
1190   P=3:D=VD:GOSUB 150  : FOR W=1 TO 1000:NEXT 'Vg and Wait
1200   GOSUB 870  'COUNTERS READ
1210   D2=FND2(P) 'ADC READ
1220   IF I=0 THEN FI#=DE#:DG#=DE#
1230   DD#=DE#-DG#:DG#=DE# 'DIFFERENCE
1240   IF MODE AND 4 THEN DE#(I)=DE#: FR#(I)=DF#: AD(I)=D2
```

```
1250    I=I+1
1260    DO#=DD#/KE#*4.096+2048
1265    IF VS=0 THEN DO#=DF#/KE#*4.096:GOTO 1270 'DAC(4)=I/Vg
1270    D4=DO#-INT(DO#/4096)*4096
1280    D=D4
1290    GOSUB 170  'DAC(4)
1300    DO#=DE#/KF#:DP#=(DE#-OF#)/KF#*4.096
1310    D1=DP#-INT(DP#/4096)*4096
1320    D=D1
1330    GOSUB 110  'DAC(1)
1340    PRINT @ 64*C, USING L4$;VD,DE#,DD#,D4,DF#,D2,D1;
1350    C=C+1: IF C=16 THEN C=USR0(C): C=15: PRINT @64*C,STRING$(63,32);
1360    IF MODE AND 8 THEN LPRINT USING L4$;VD,DE#,DD#,D4,DF#,D2,D1
1362    IF MODE AND 48 THEN 1364 ELSE 1370
1364      T=PEEK(A)+PEEK(A+1)*60+PEEK(A+2)*3600
1368      IF T>T0 THEN IF MODE AND 16 THEN LPRINT USING "########  "+L4$;T,VD,DE#
,DD#,D4,DF#,D2,D1:T0=T0+TS:ELSE PRINT #1,T;DE#;DF#;D2:T0=T0+TS
1370    IF I$="S" THEN 1390 ELSE I$=INKEY$
1380 NEXT VD
1390 '-- END OF MEASUREMENT
1400 PRINT STRING$(4,27):MODE=MODE OR 2:UA=I:I$=""
1410 PRINT "END OF MEASUREMENT"
1412 PRINT "INVERSE (Y/N) <";A$;">";:INPUT A$
1414 IF A$="Y" THEN EN=ST:ST=VD:VS=-VS:GOTO 1180
1416 A$="Y"
1420 PRINT "VOLT ZERO (Y/S/N) <";A$;">";:INPUT A$
1430 IF A$="Y" THEN PRINT CHR$(27);STRING$(30,25);:DE=0: P=3: GOSUB 520:PRINT:
ELSE IF A$="S" THEN D=0:GOSUB 150
1435 IF (MODE AND 4)=0 THEN 1500 'If there are no DATA
1440 A$="N":PRINT "X-t Recorder OUT(Y/N) <";A$;">";:INPUT A$
1450 IF A$="Y" THEN GOSUB 1520
1460 PRINT "PRINTER OUT(Y/N) <";A$;">";:INPUT A$
1470 IF A$="Y" THEN GOSUB 1760
1480 PRINT "DISK WRITE(Y/N) <";A$;">";:INPUT A$
1490 IF A$="Y" THEN GOSUB 1930
1500 P=10:CLOSE:RETURN
1510 END
1520 '-- X-t Recorder OUT --
1530 INPUT "TIME INTERVAL (mS) ";W1
1540 INPUT "DAC(4) ｶﾗ X-t Recorder I ﾀﾞｽ ﾊﾞｲﾘﾂ d(Vd/Vg)/2↑N, N=";J
1550 IF J>0 THEN KE#=.5#:FOR I=0 TO J:KE#=KE#*2:NEXT I ELSE KE#=2#:FOR I=J TO 0:
KE#=KE#/2:NEXT I
1560 INPUT "DAC(1) ｶﾗ X-t Recorder I ﾀﾞｽ ﾊﾞｲﾘﾂ Frequency/2↑N,N=";J
1570 IF J>0 THEN KF#=.5#:FOR I=0 TO J:KF#=KF#*2:NEXT I ELSE KF#=2#:FOR I=J TO 0:
KF#=KF#/2:NEXT I
1580 I=0
1590 FOR VD=ST TO EN STEP VS
1600    '- DAC(3) - ｲﾙﾉｶﾅ ?
1610    'GOSUB 30
1620    IF I=0 THEN DD#=0 ELSE DD#=DE#(I+1)-DE#(I)
1630    DO#=DD#/KE#+2048
1640    D4=DO#-INT(DO#/4096)*4096
1650    '- DAC(4) -
1660    D=D4:GOSUB 170
1670    DO#=DE#(I)/KF#+2048
1680    D1=DO#-INT(DO#/4096)*4096
1690    '- DAC(1) -
1700    D=D1:GOSUB 110
1710    FOR W=0 TO W1: NEXT '- WAIT -
1720    I=I+1:PRINT USING F$;I;:IF I=UA THEN VD=EN
1730 NEXT VD
1740 PRINT: PRINT "X-t OUT END"
1750 RETURN
```

```
1760 '-- PRINTER OUT --
1770 LINEINPUT "Comment ? ";CM$
1780 LPRINT :LPRINT CM$: LPRINT L3$
1790 I=0
1800 FOR VD=ST TO EN STEP VS
1810   DE#=DE#(I)
1820   IF I=0 THEN DD#=0 ELSE DD#=DE#(I+1)-DE#(I)
1830   DO#=DE#/KF#
1840   D4=DO#-INT(DO#/4096)*4096
1850   DF#=FR#(I)
1860   DO#=DF#/KF#+2048
1870   D1=DO#-INT(DO#/4096)*4096
1880   LPRINT USING L4$;VD,DE#,DD#,D4,DE#,D2,D1
1890   I=I+1:IF I=UA THEN VD=EN
1900 NEXT VD
1910 RETURN
1930 ' -- WRITE DISK FILE --
1940 LINEINPUT "DISK FILE NAME ? ";A$
1950 LINEINPUT "COMMENT ? ";CM$
1960 OPEN "O",1,A$
1970 PRINT #1,CM$
1980 PRINT #1,ST;EN;VS;UA
1990 FOR I=0 TO UA
2000   PRINT #1,DE#(I);FR#(I);AD(I);
2010 NEXT I
2020 CLOSE
2030 PRINT "DISK WRITE END "
2040 RETURN
2050 END
3000 LINEINPUT "FILENAME ? ";A$
3010 OPEN "I",1,A$
3020 LINEINPUT #1,A$:PRINT A$
3030 INPUT #1,ST,EN,VS,UA:PRINT ST,EN,VS,UA
3040 FOR I=0 TO UA
3050   INPUT #1,DE#,DF#,D2:PRINT DE#,DF#,D2
3060 NEXT I
3070 CLOSE
```

図 10. 内部摩擦測定ソフト。

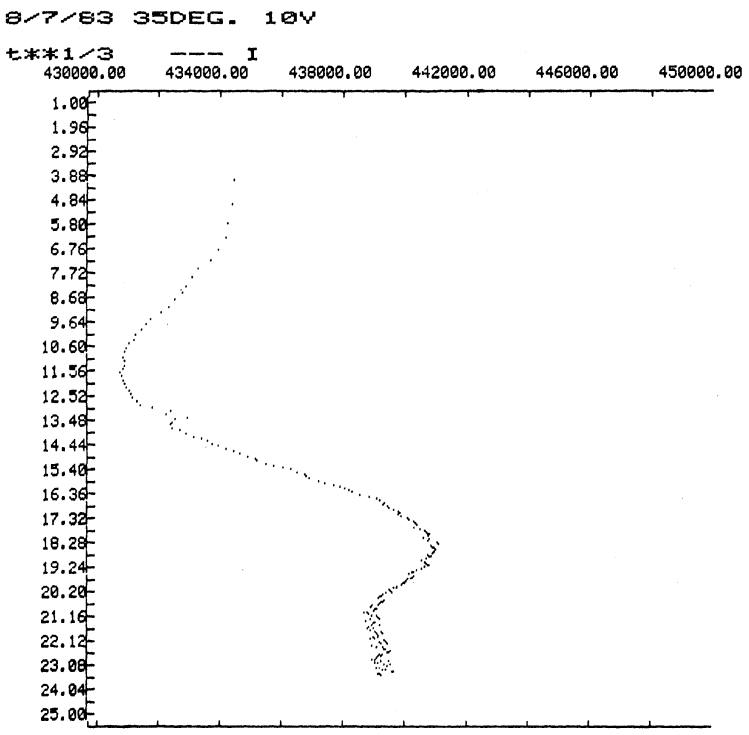

図 11. 内部摩擦の測定例。